CLAIMS

What is claimed is:

1. A method of managing memory, comprising:

examining current and future instructions operating on a stack;

determining stack trend information; and

utilizing the trend information to reduce data traffic between various levels of a memory.

2. The method of claim 1, wherein determining the trend information includes examining future instructions to determine if the size of the stack is going to decrease as a result of future instructions.

3. The method of claim 2, wherein a predetermined number of instructions are used in determining stack trend information.

4. The method of claim 3, wherein the number of predetermined instructions is at least two.

5. The method of claim 3, wherein the cache memory maintains a single dirty cache line for stack data.

6. The method of claim 3, wherein if a dirty cache line needs to be written back, then analyzing the trend information, which includes determining which word of the dirty cache line is going to be written to.

7. The method of claim 1, wherein the levels of memory comprise a cache memory containing multiple cache lines and a main memory, and wherein the trend information is used to restrict writing dirty cache lines from cache memory to main memory when the trend information indicates the stack is decreasing.

8. The method of claim 1, wherein determining the trend information includes examining future instructions to determine if the size of the stack is going to increase as a result of future instructions.

9. The method of claim 8, wherein determining if a line is written back includes analyzing the trend information and includes examining a dirty cache line to determine which word of the dirty cache line is going to be written to.

10. The method of claim 9, wherein the dirty cache line is written from a cache memory to a main memory.

11. A computer system, comprising:

a processor;

a memory coupled to the processor;

a stack that exists in memory and contains stack data;

a memory controller coupled to the memory;

trend logic;

wherein the processor executes instructions;

wherein the trend logic provides trend information about the stack to the controller; and

wherein the trend information about the stack is based on at least one future instruction.

12. The computer system of claim 11, further comprising an instruction decoder comprising a first portion that decodes current instructions and a second portion that decodes future instructions.

13. The computer system of claim 12, wherein the trend logic determines a net stack trend based on current instruction and future instruction information coming from the decode logic.

14. The computer system of claim 12, wherein the second potion of the decoder is adjusted so that the number of future instructions that are decoded equals at least two.

15. The computer system of claim 11, wherein the memory further includes a cache memory containing multiple cache lines and a main memory, and wherein the trend information is used to restrict writing dirty cache lines from cache memory to main memory when the trend information indicates the stack is decreasing.

16. The computer system of claim 11, wherein the memory further includes a cache memory and a main memory, and wherein the cache memory contains a dirty cache line, and wherein the dirty cache line is written to main memory if the trend information indicates the stack is increasing.

17. A method, comprising:

issuing a write request to a cache memory, wherein the cache memory includes multiple cache lines;

determining whether the write request refers to a predetermined word within a dirty cache line; and

determining whether the size of a stack is increasing or decreasing.

18. The method of claim 17, further comprising determining whether the write request will be to the end of a dirty cache line.

19. The method of claim 18, wherein the stack size is increasing and the dirty cache line is written to a main memory.

20. The method of claim 18, wherein the stack size decreasing and the dirty cache line is retained in the cache memory.